Parallel Simulation of High Reynolds Number Vascular Flows

Paul Fischer,[a] Francis Loth, Sang-Wook Lee, David Smith, Henry Tufo, and Hisham Bassiouny

[a]Mathematics and Computer Science Division, Argonne National Laboratory
Argonne, IL 60439, U.S.A.

1. Introduction

The simulation of turbulent vascular flows presents significant numerical challenges. Because such flows are only weakly turbulent (i.e., transitional), they lack an inertial subrange that is amenable to subgrid-scale (SGS) modeling required for large-eddy or Reynolds-averaged Navier-Stokes simulations. The only reliable approach at present is to directly resolve all scales of motion. While the Reynolds number is not high (Re=1000–2000, typ.), the physical dissipation is small, and high-order methods are essential for efficiency. Weakly turbulent blood flow, such as occurs in post-stenotic regions or subsequent to graft implantation, exhibits a much broader range of scales than does its laminar (healthy) counterpart and thus requires an order of magnitude increase in spatial and temporal resolution, making fast iterative solvers and parallel computing necessities.

The paper is organized as follows. Section 2 provides a brief overview of the governing equations, time advancement scheme, and spectral element method. Section 3 describes boundary condition treatment for simulating transition in bifurcation geometries. Section 4 presents parallel considerations and performance results, and Section 5 gives results for transitional flow in an arteriovenous graft model.

2. Navier-Stokes Discretization

We consider the solution of incompressible Navier-Stokes equations in Ω,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} = -\nabla p + \frac{1}{Re}\nabla^2\mathbf{u}, \qquad \nabla\cdot\mathbf{u} = 0, \tag{1}$$

subject to appropriate initial and boundary conditions. Here, $\mathbf{u}$ is the velocity field, p is the pressure normalized by the density, and $Re = UD/\nu$ is the Reynolds number based on the characteristic velocity U, length scale D, and kinematic viscosity ν.

Our temporal discretization is based on a semi-implicit formulation in which the nonlinear terms are treated explicitly and the remaining linear Stokes problem is treated implicitly. We approximate the time derivative in (1) using a kth-order backwards difference formula (BDFk, k=2 or 3), which for k=2 reads

$$\frac{3\mathbf{u}^n - 4\mathbf{u}^{n-1} + \mathbf{u}^{n-2}}{2\Delta t} = S(\mathbf{u}^n) + NL^n. \tag{2}$$

Here, $\mathbf{u}^{n-q}$ represents the velocity at time t^{n-q}, $q = 0, \ldots, 2$, and $S(\mathbf{u}^n)$ is the linear symmetric Stokes operator that implicitly incorporates the divergence-free constraint. The term NL^n approximates the nonlinear terms at time level t^n and is given by the extrapolant $NL^n := -\sum_j \alpha_j \mathbf{u}^{n-j} \cdot \nabla \mathbf{u}^{n-j}$. For $k = 2$, the standard extrapolation would use $\alpha_1 = 2$ and $\alpha_2 = -1$. Typically, however, we use the three-term second-order formulation with $\alpha_1 = 8/3$, $\alpha_2 = -7/3$, and $\alpha_3 = 2/3$, which has a stability region that encompasses part of the imaginary axis. As an alternative to (2), we frequently use the operator-integration-factor scheme of Maday et al. [10] that circumvents the CFL stability constraints by setting $NL^n = 0$ and replacing the left-hand side of (2) with an approximation to the material derivative of $\mathbf{u}$. Both formulations yield unsteady Stokes problems of the form

$$\begin{array}{lcl} \mathcal{H}\mathbf{u}^n & - \;\nabla p^n & = \;\mathbf{f}^n \\ \nabla \cdot \mathbf{u}^n & & = \;0, \end{array} \tag{3}$$

to be solved implicitly. Here, $\mathcal{H}$ is the Helmholtz operator $\mathcal{H} := \left(\frac{3}{2\Delta t} - \frac{1}{Re}\nabla^2\right)$. In Section 3, we will formally refer to (3) in operator form $S_{us}(\mathbf{u}^n) = \mathbf{f}^n$. In concluding our temporal discretization overview, we note that we often stabilize high-Re cases by filtering the velocity at each step ($\mathbf{u}^n = F(\mathbf{u}^n)$), using the high-order filter described in [3,5].

Our spatial discretization of (3) is based on the $\mathbb{P}_N - \mathbb{P}_{N-2}$ spectral element method (SEM) of Maday and Patera [9]. The SEM is a high-order weighted residual approach similar to the finite element method (FEM). The primary distinction between the two is that typical polynomial orders for the SEM bases are in the range N=4 to 16—much higher than for the FEM. These high orders lead to excellent transport (minimal numerical diffusion and dispersion) for a much larger fraction of the resolved modes than is possible with the FEM. The relatively high polynomial degree of the SEM is enabled by the use of tensor-product bases having the form (in 2D)

$$\mathbf{u}(\mathbf{x}^e(r,s))|_{\Omega^e} \;=\; \sum_{i=0}^{N}\sum_{j=0}^{N} \mathbf{u}^e_{ij} h^N_i(r) h^N_j(s)\,, \tag{4}$$

which implies the use of (curvilinear) quadrilateral (2D) or hexahedral (3D) elements. Here, $\mathbf{u}^e_{ij}$ is the nodal basis coefficient on element Ω^e; $h^N_i \in \mathbb{P}_N$ is the Lagrange polynomial based on the Gauss-Lobatto quadrature points, $\{\xi^N_j\}^N_{j=0}$ (the zeros of $(1-\xi^2)L'_N(\xi)$, where L_N is the Legendre polynomial of degree N); and $\mathbf{x}^e(r,s)$ is the coordinate mapping from $\hat{\Omega} = [-1,1]^d$ to Ω^e, for d=2 or 3. Unstructured data accesses are required at the global level (i.e., $e = 1, ..., E$), but the data is accessed in i-j-k form within each element. In particular, differentiation—a central kernel in operator evaluation—can be implemented as a cache-efficient matrix-matrix product. For example, $\mathbf{u}_{r,ij} = \sum_p \widehat{D}_{ip}\mathbf{u}_{pj}$, with $\widehat{D}_{ip} := h'_p(\xi_i)$ would return the derivative of (4) with respect to the computational coordinate r at the points (ξ_i, ξ_j). Differentiation with respect to $\mathbf{x}$ is obtained by the chain rule [1].

Insertion of the SEM basis (4) into the weak form of (3) and applying numerical quadrature yields the discrete unsteady Stokes system

$$H\,\underline{\mathbf{u}}^n - D^T\underline{p}^n = B\,\underline{\mathbf{f}}^n, \;\; D\,\underline{\mathbf{u}}^n = 0. \tag{5}$$

Here, $H = \frac{1}{Re}A + \frac{3}{2\Delta t}B$ is the discrete equivalent of $\mathcal{H}$; $-A$ is the discrete Laplacian, B is the (diagonal) mass matrix associated with the velocity mesh, D is the discrete divergence operator, and $\underline{\mathbf{f}}^n$ accounts for the explicit treatment of the nonlinear terms.

The Stokes system (5) is solved approximately, using the kth-order operator splitting analyzed in [10]. The splitting is applied to the *discretized* system so that ad hoc boundary conditions are avoided. For $k = 2$, one first solves

$$H\,\underline{\hat{\mathbf{u}}} = B\,\underline{\mathbf{f}}^n + D^T\,\underline{p}^{n-1}, \tag{6}$$

which is followed by a pressure correction step

$$E\delta\underline{p} = -D\underline{\hat{\mathbf{u}}}, \qquad \underline{\mathbf{u}}^n = \underline{\hat{\mathbf{u}}} + \Delta t B^{-1} D^T \delta\underline{p}, \qquad \underline{p}^n = \underline{p}^{n-1} + \delta\underline{p}, \tag{7}$$

where $E := \frac{2}{3}\Delta t D B^{-1} D^T$ is the Stokes Schur complement governing the pressure in the absence of the viscous term. Substeps (6) and (7) are solved with preconditioned conjugate gradient (PCG) iteration. Jacobi preconditioning is sufficient for (6) because H is strongly diagonally dominant. E is less well-conditioned and is solved either by the multilevel overlapping Schwarz method developed in [2,4] or more recent Schwarz-multigrid methods [8].

3. Boundary Conditions

Boundary conditions for the simulation of transition in vascular flow models present several challenges not found in classical turbulence simulations. As velocity profiles are rarely available, our usual approach at the vessel inflow is to specify a time-dependent Womersely flow that matches the first 20 Fourier harmonics of measured flow waveform. In some cases, it may be necessary to augment such clean profiles with noise in order to trigger transition at the Reynolds numbers observed in vivo. At the outflow, our standard approach is to use the natural boundary conditions (effectively, $p = 0$ and $\frac{\partial \mathbf{u}}{\partial n} = 0$) associated with the variational formulation of (3). This outflow boundary treatment is augmented in two ways for transitional vascular flows, as we now describe.

Fast Implicit Enforcement of Flow Division. Imposition of proper flow division (or flow split) is central to accurate simulation of vascular flows through bifurcations (sites prone to atherogenesis). The distribution of volumetric flow rate through multiple daughter branches is usually available through measured volume flow rates. A typical distribution in a carotid artery bifurcation, for example, is a 60:40 split between the internal and external carotid arteries. The distribution can be time-dependent, and the method we outline below is applicable to such cases. A common approach to imposing a prescribed flow split is to apply Dirichlet velocity conditions at one outlet and standard outflow (Neumann) conditions at the other. The Dirichlet branch is typically artificially extended to diminish the influence of spurious boundary effects on the upstream region of interest. Here, we present an approach to imposing arbitrary flow divisions among multiple branches that allows one to use Neumann conditions at each of the branches, thus reducing the need for extraordinary extensions of the daughter branches.

Our flow-split scheme exploits the semi-implicit approach outlined in the preceding section. The key observation is that the unsteady Stokes operator, which is treated implicitly and which controls the boundary conditions, is *linear* and that superposition therefore applies. Thus, if $\tilde{\mathbf{u}}^n$ satisfies $S_{us}(\tilde{\mathbf{u}}^n) = \mathbf{f}^n$ and $\tilde{\mathbf{u}}_0$ satisfies $S_{us}(\tilde{\mathbf{u}}_0) = 0$ but with different boundary conditions, then $\mathbf{u}^n := \tilde{\mathbf{u}}^n + \tilde{\mathbf{u}}_0$ will satisfy $S_{us}(\mathbf{u}^n + \tilde{\mathbf{u}}_0) = \mathbf{f}^n$

with boundary conditions $\mathbf{u}^n|_{\partial\Omega} = \tilde{\mathbf{u}}^n|_{\partial\Omega} + \tilde{\mathbf{u}}_0|_{\partial\Omega}$. With this principle, the flow split for a simple bifurcation (one common inflow, two daughter outflow branches) is imposed as follows. In a preprocessing step:

(i) Solve $S_{us}(\tilde{\mathbf{u}}_0) = 0$ with a prescribed inlet profile having flux $\tilde{Q} := \int_{\text{inlet}} \tilde{\mathbf{u}}_0 \cdot \mathbf{n}\, dA$, and no flow (i.e., homogeneous Dirichlet conditions) at the exit of one of the daughter branches. Use Neumann (natural) boundary conditions at the the other branch. Save the resultant velocity-pressure pair $(\tilde{\mathbf{u}}_0, \tilde{p}_0)$.

(ii) Repeat the above procedure with the role of the daughter branches reversed, and call the solution $(\tilde{\mathbf{u}}_1, \tilde{p}_1)$.

Then, at each timestep:

(iii) Compute $(\tilde{\mathbf{u}}^n, \tilde{p}^n)$ satisfying (3) with homogeneous Neumann conditions on each daughter branch and compute the associated fluxes $\tilde{Q}_i^n := \int_{\partial\Omega_i} \tilde{\mathbf{u}}^n \cdot \mathbf{n}\, dA$, i=0, 1, where $\partial\Omega_0$ and $\partial\Omega_1$ are the respective active exits in (*i*) and (*ii*) above.

(iv) Solve the following for (α_0, α_1) to obtain the desired flow split Q_0^n:Q_1^n:

$$Q_0^n = \tilde{Q}_0^n + \alpha_0 \tilde{Q} \qquad \text{(desired flux on branch 0)} \tag{8}$$

$$Q_1^n = \tilde{Q}_1^n + \alpha_1 \tilde{Q} \qquad \text{(desired flux on branch 1)} \tag{9}$$

$$0 = \alpha_0 + \alpha_1 \qquad \text{(change in flux at inlet)} \tag{10}$$

(v) Correct the solution by setting $\mathbf{u}^n := \tilde{\mathbf{u}}^n + \sum_i \alpha_i \tilde{\mathbf{u}}_i$ and $p^n := \tilde{p}^n + \sum_i \alpha_i \tilde{p}_i$.

Remarks. The above procedure provides a fully implicit iteration-free approach to applying the flow split that readily extends to a larger number of branches by expanding the system (8)–(10). Condition (10) ensures that the net flux at the inlet is unchanged and, for a simple bifurcation, one only needs to store the difference between the auxiliary solutions. We note that S_{us} is dependent on the timestep size Δt and that the auxiliary solutions $(\tilde{\mathbf{u}}_i, \tilde{p}_i)$ must be recomputed if Δt (or ν) changes. The amount of viscous diffusion that can take place in a single application of the unsteady Stokes operator is governed by Δt, and one finds that the auxiliary solutions have relatively thin boundary layers with a broad flat core. The intermediate solutions obtained in (*iii*) have inertia and so nearly retain the proper flow split, once established, such that the magnitude of α_i will be relatively small after just a few timesteps. It is usually a good idea to gradually ramp up application of the correction if the initial condition is not near the desired flow split. Otherwise, one runs the risk of having reversed flow on portions of the outflow boundary and subsequent instability, as discussed in the next section. Moreover, to accommodate the exit "nozzle" ($\nabla \cdot \mathbf{u} > 0$) condition introduced below, which changes the net flux out of the exit, we compute $\tilde{Q}_i^n$ at an upstream cross-section where $\nabla \cdot \mathbf{u} = 0$.

Turbulent Outflow Boundary Conditions. In turbulent flows, it is possible to have vortices of sufficient strength to yield a (locally) negative flux at the outflow boundary. Because the Neumann boundary condition does not specify flow characteristics at the exit, a negative flux condition can rapidly lead to instability, with catastrophic results. One way to eliminate incoming characteristics is to force the exit flow through a nozzle, effectively adding a mean axial component to the velocity field. The advantage of using a nozzle is that one can ensure that the characteristics at the exit point outward under a wide range of flow conditions. By constrast, schemes based on viscous buffer zones require knowledge of the anticipated space and time scales to ensure that vortical structures are adequately damped as they pass through the buffer zone.

Numerically, a nozzle can be imposed without change to the mesh geometry by imparting a positive divergence to the flow field near the exit (in the spirit of a supersonic nozzle). In our simulations, we identify the layer of elements adjacent to the outflow and there impose a divergence $D(\mathbf{x})$ that ramps from zero at the upstream end of the layer to a fixed positive value at the exit. Specifically, we set $D(\mathbf{x}) = C[\,1-(x_\perp/L_\perp)^2\,]$, where $x_\perp$ is the distance normal to the boundary and $L_\perp$ is maximum thickness of the last layer of elements. By integrating the expression for D from $x_\perp/L_\perp$=1 to 0, one obtains the net gain in mean velocity over the extent of the layer. We typically choose the constant C such that the gain is equal to the mean velocity prior to the correction.

Results for the nozzle-based outflow condition are illustrated in Fig. 1. The left panel shows the velocity field for the standard (uncorrected Neumann) condition near the outflow boundary of an internal carotid artery at $Re \approx 1400$ (based on the peak flow rate and stenosis diameter). Inward-pointing velocity vectors can be seen at the exit boundary, and the simulation becomes catastrophically unstable within 100 timesteps beyond this point. The center panel shows the flow field computed with the outflow correction. The flow is leaving the domain at all points along the outflow boundary and the simulation is stable for all time. The difference between the two cases (right) shows that the outflow treatment does not pollute the solution upstream of the boundary.

4. Parallel Performance

Our approach to parallelization is based on standard SPMD domain decomposition approaches, as discussed in [1,13]. Elements are distributed across processors in contiguous subgroups determined by recursive spectral bisection [11] and nearest neighbor data is exchanged with each matrix-vector product required for the iterative solvers. (See [1].)

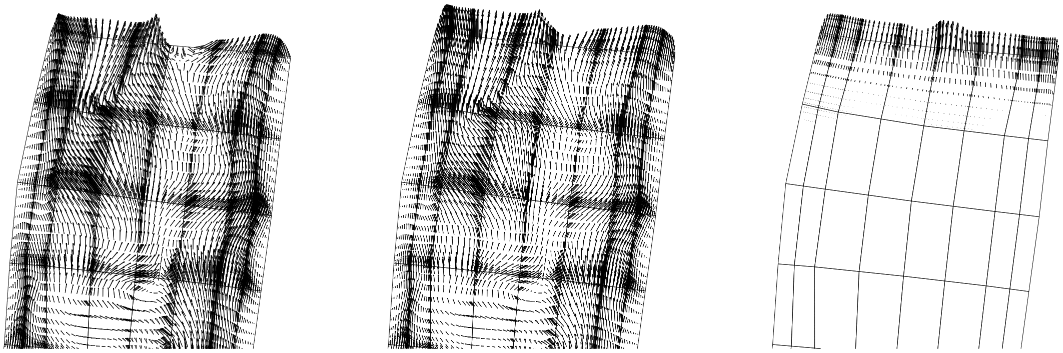

Figure 1. Velocity vectors near the outflow of an internal carotid artery: (left) uncorrected, (center) corrected, and (right) corrected-uncorrected.

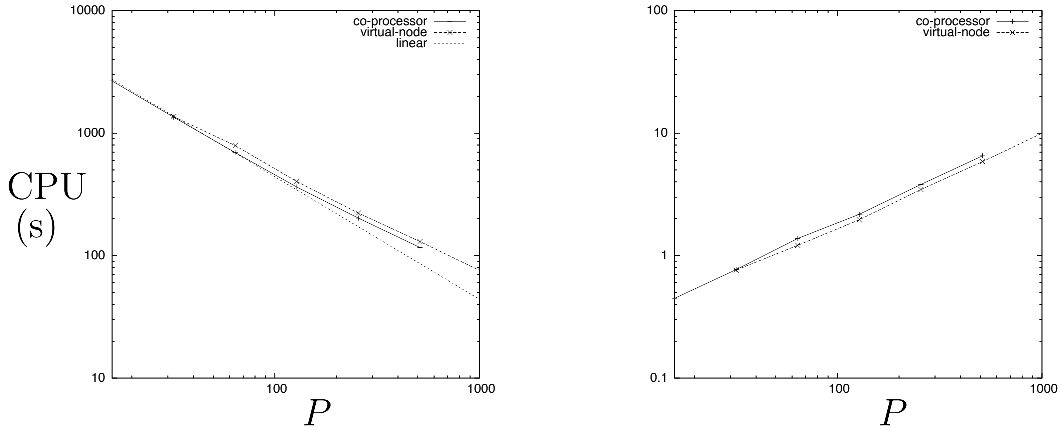

Figure 2. (left) CPU time for E=2640, N=10 for P=16–1024 using coprocessor and virtual-node modes on BGL; (right) percentage of time spent in the coarse-grid solve.

The only other significant communication arises from inner products in the PCG iteration and from the coarse-grid problem associated with the pressure solve.

The development of a fast coarse-grid solver for the multilevel pressure solver was central to efficient scaling to $P > 256$ processors. The use of a coarse-grid problem in multigrid and Schwarz-based preconditioners ensures that the iteration count is bounded independent of the number of subdomains (elements) [12]. The coarse-grid problem, however, is communication intensive and generally not scalable. We employ the projection-based coarse-grid solver developed in [14]. This approach has a communication complexity that is sublinear in P, which is a significant improvement over alternative approaches, which typically have communication complexities of $O(P \log P)$.

Figure 2 (left) shows the CPU time vs. P on the IBM BGL machine at Argonne for 50 timesteps for the configuration of Fig. 3. For the P=1024 case, the parallel efficiency is $\eta = .56$, which is respectable for this relatively small problem ($\approx$ 2600 points/processor). The percentage of time spent in the coarse-grid solver is seen in Fig. 2 (right) to be less than 10 percent for $P \leq 1024$. BGL has two processors per node that can be used in coprocessor (CO) mode (the second processor handles communication) or in virtual-node (VN) mode (the second processor is used for computation). Typically about a 10% overhead is associated with VN-mode. For example, for $P = 512$, we attain $\eta = .72$ in CO-mode and only $\eta = .64$ in VN-mode. Obviously, for a given P, VN-mode uses half as many resources and is to be preferred.

5. Transition in an Arteriovenous Graft

Arteriovenous (AV) grafts consist of a ∼15 cm section of 6 mm i.d. synthetic tubing that is surgically implanted to provide an arterial-to-vein round-the-clock short circuit. Because they connect a high-pressure vessel to a low-pressure one, high flow rates are established that make AV-grafts efficient dialysis ports for patients suffering from poor kidney function. The high speed flow is normally accompanied by transition to a weakly turbulent state, manifested as a 200–300 Hz vibration at the vein wall [6,7]. This high-

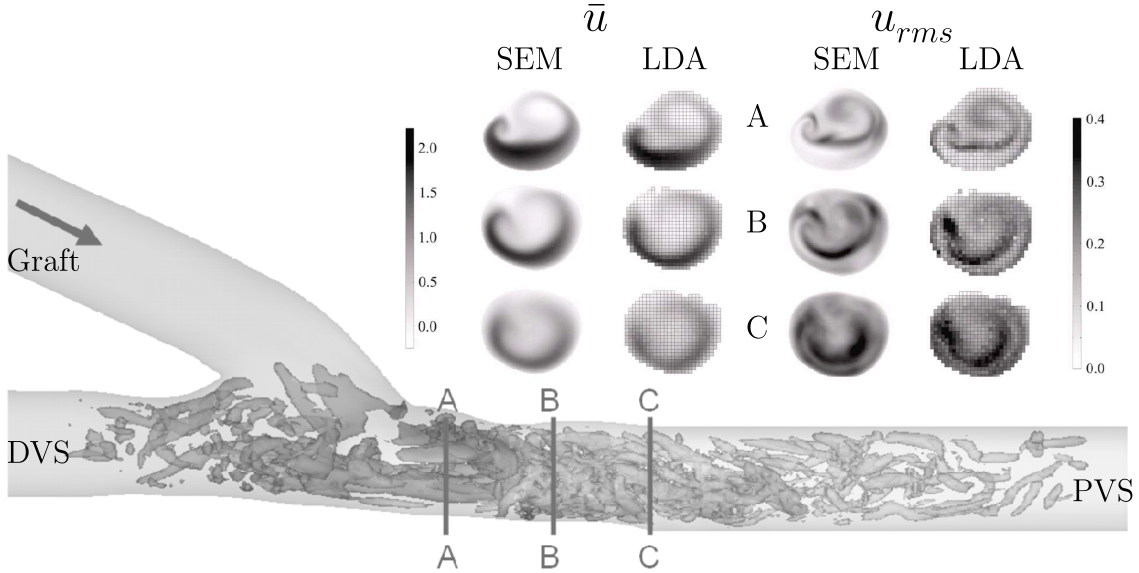

Figure 3. Transitional flow in an AV graft at $Re = 1200$: (bottom) coherent structures; (inset) mean and rms velocity distributions (m/s) at A–C for CFD and LDA measurements. The mesh comprized 2640 elements of order $N = 12$ with $\Delta t = 5 \times 10^6 s$.

frequency excitation is thought to lead to intimal hyperplasia, which can lead to complete occlusion of the vein and graft failure within six months of implant. We are currently investigating the mechanisms leading to transition in subject-specific AV-graft models with the aim of reducing turbulence through improved geometries. Figure 3 shows a typical turbulent case when there is a 70:30 split between the proximal venous segment (PVS) and distal venous segment (DVS). The SEM results, computed with 2640 elements of order 12 (4.5 million gridpoints), are in excellent agreement with concurrent laser Doppler anemometry results. The statistics are based on 0.5 seconds of (in vivo) flow time, which takes about 100,000 steps and 20 hours of CPU time on 2048 processors of BGL.

Acknowledgments

This work was supported by the National Institutes of Health, RO1 Research Project Grant (2RO1HL55296-04A2), by Whitaker Foundation Grant (RG-01-0198), and by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

REFERENCES

1. M.O. Deville, P.F. Fischer, and E.H. Mund, *High-order methods for incompressible fluid flow*, Cambridge University Press, Cambridge, 2002.
2. P.F. Fischer, *An overlapping Schwarz method for spectral element solution of the incompressible Navier-Stokes equations*, J. Comput. Phys. **133** (1997), 84–101.

3. P.F. Fischer, G.W. Kruse, and F. Loth, *Spectral element methods for transitional flows in complex geometries*, J. Sci. Comput. **17** (2002), 81–98.
4. P.F. Fischer, N.I. Miller, and H.M. Tufo, *An overlapping Schwarz method for spectral element simulation of three-dimensional incompressible flows*, Parallel Solution of Partial Differential Equations (Berlin) (P. Bjørstad and M. Luskin, eds.), Springer, 2000, pp. 158–180.
5. P.F. Fischer and J.S. Mullen, *Filter-based stabilization of spectral element methods*, Comptes rendus de l'Académie des sciences, Série I- Analyse numérique **332** (2001), 265–270.
6. S.W. Lee, P.F. Fischer, F. Loth, T.J. Royston, J.K. Grogan, and H.S. Bassiouny, *Flow-induced vein-wall vibration in an arteriovenous graft*, J. of Fluids and Structures (2005).
7. F. Loth, N. Arslan, P. F. Fischer, C. D. Bertram, S. E. Lee, T. J. Royston, R. H. Song, W. E. Shaalan, and H. S. Bassiouny, *Transitional flow at the venous anastomosis of an arteriovenous graft: Potential relationship with activation of the ERK1/2 mechanotransduction pathway*, ASME J. Biomech. Engr. **125** (2003), 49–61.
8. J. W. Lottes and P. F. Fischer, *Hybrid multigrid/Schwarz algorithms for the spectral element method*, J. Sci. Comput. **24** (2005).
9. Y. Maday and A.T. Patera, *Spectral element methods for the Navier-Stokes equations*, State-of-the-Art Surveys in Computational Mechanics (A.K. Noor and J.T. Oden, eds.), ASME, New York, 1989, pp. 71–143.
10. Y. Maday, A.T. Patera, and E.M. Rønquist, *An operator-integration-factor splitting method for time-dependent problems: Application to incompressible fluid flow*, J. Sci. Comput. **5** (1990), 263–292.
11. A. Pothen, H.D. Simon, and K.P. Liou, *Partitioning sparse matrices with eigenvectors of graphs*, SIAM J. Matrix Anal. Appl. **11** (1990), 430–452.
12. B. Smith, P. Bjørstad, and W. Gropp, *Domain decomposition: Parallel multilevel methods for elliptic PDEs*, Cambridge University Press, Cambridge, 1996.
13. H.M. Tufo and P.F. Fischer, *Terascale spectral element algorithms and implementations*, Proc. of the ACM/IEEE SC99 Conf. on High Performance Networking and Computing (IEEE Computer Soc.), 1999, p. CDROM.
14. ———, *Fast parallel direct solvers for coarse-grid problems*, J. Parallel Distrib. Comput. **61** (2001), 151–177.

The submitted manuscript has been created by the University of Chicago as Operator of Argonne National Laboratory ("Argonne") under Contract No. W-31-109-ENG-38 with the U.S. Department of Energy. The U.S. Government retains for itself, and others acting on its behalf, a paid-up, nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.